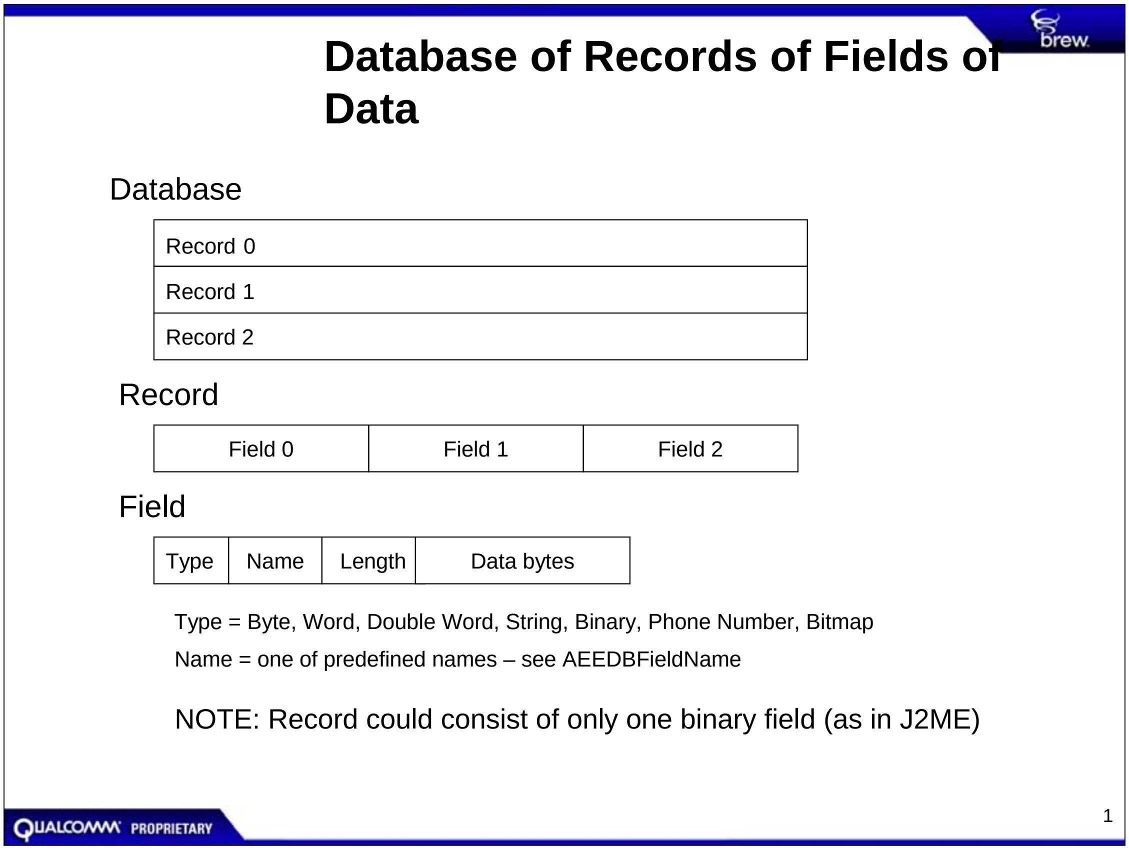


Database of Records of Fields of Data

Database

Record 0
Record 1
Record 2

Record

Field 0	Field 1	Field 2

Field

Type	Name	Length	Data bytes

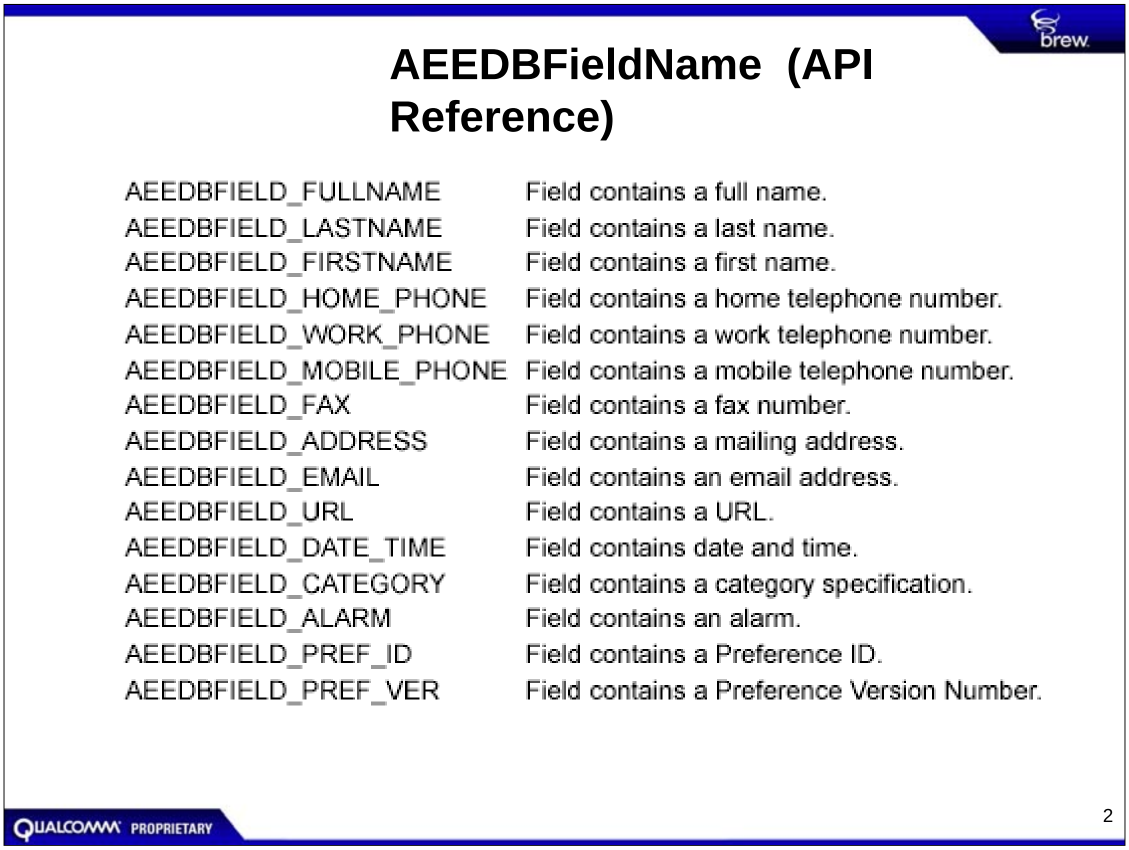
Type = Byte, Word, Double Word, String, Binary, Phone Number, Bitmap

Name = one of predefined names – see AEEDBFieldName

NOTE: Record could consist of only one binary field (as in J2ME)

QUALCOMM PROPRIETARY

1

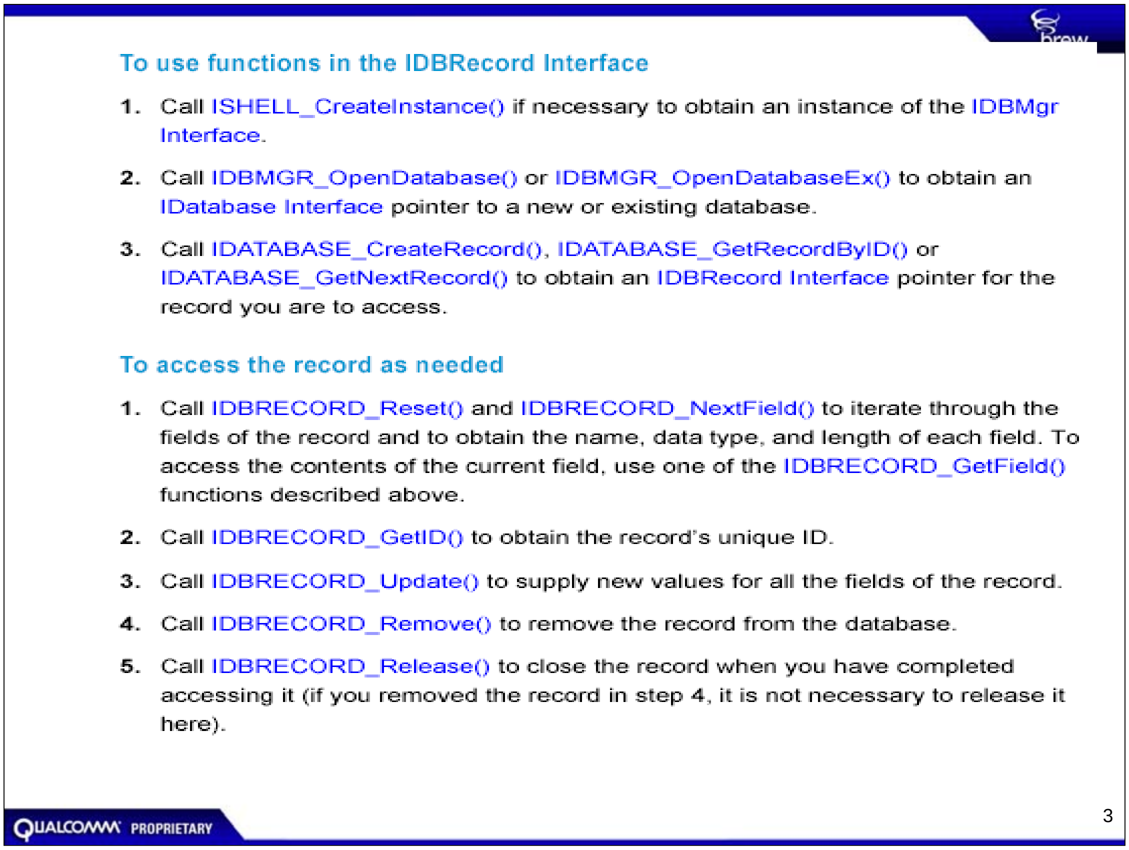


AEEDBFieldName (API Reference)

AEEDBFIELD_FULLNAME	Field contains a full name.
AEEDBFIELD_LASTNAME	Field contains a last name.
AEEDBFIELD_FIRSTNAME	Field contains a first name.
AEEDBFIELD_HOME_PHONE	Field contains a home telephone number.
AEEDBFIELD_WORK_PHONE	Field contains a work telephone number.
AEEDBFIELD_MOBILE_PHONE	Field contains a mobile telephone number.
AEEDBFIELD_FAX	Field contains a fax number.
AEEDBFIELD_ADDRESS	Field contains a mailing address.
AEEDBFIELD_EMAIL	Field contains an email address.
AEEDBFIELD_URL	Field contains a URL.
AEEDBFIELD_DATE_TIME	Field contains date and time.
AEEDBFIELD_CATEGORY	Field contains a category specification.
AEEDBFIELD_ALARM	Field contains an alarm.
AEEDBFIELD_PREF_ID	Field contains a Preference ID.
AEEDBFIELD_PREF_VER	Field contains a Preference Version Number.

QUALCOMM PROPRIETARY

2



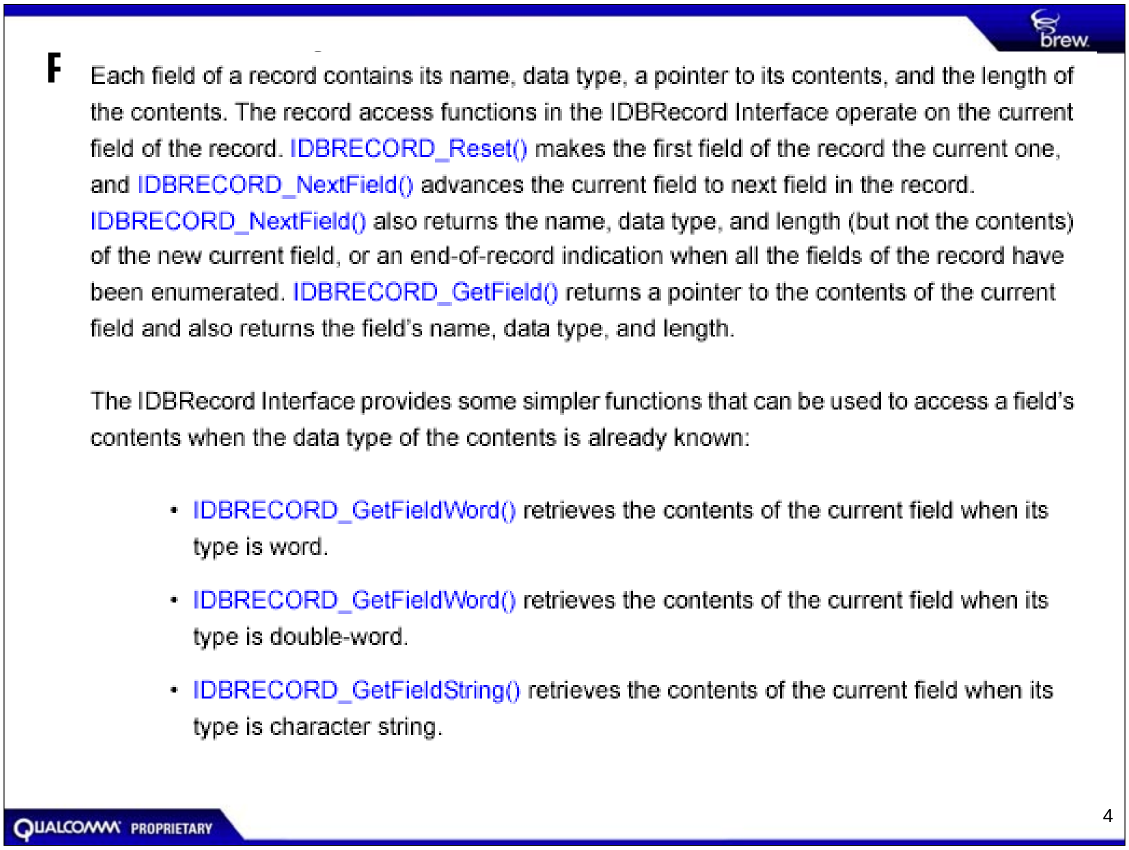
To use functions in the IDBRecord Interface

1. Call ISHELL_CreateInstance() if necessary to obtain an instance of the IDBMgr Interface.
2. Call IDBMGR_OpenDatabase() or IDBMGR_OpenDatabaseEx() to obtain an IDatabase Interface pointer to a new or existing database.
3. Call IDATABASE_CreateRecord(), IDATABASE_GetRecordByID() or IDATABASE_GetNextRecord() to obtain an IDBRecord Interface pointer for the record you are to access.

To access the record as needed

1. Call IDBRECORD_Reset() and IDBRECORD_NextField() to iterate through the fields of the record and to obtain the name, data type, and length of each field. To access the contents of the current field, use one of the IDBRECORD_GetField() functions described above.
2. Call IDBRECORD_GetID() to obtain the record's unique ID.
3. Call IDBRECORD_Update() to supply new values for all the fields of the record.
4. Call IDBRECORD_Remove() to remove the record from the database.
5. Call IDBRECORD_Release() to close the record when you have completed accessing it (if you removed the record in step 4, it is not necessary to release it here).

QUALCOMM PROPRIETARY 3

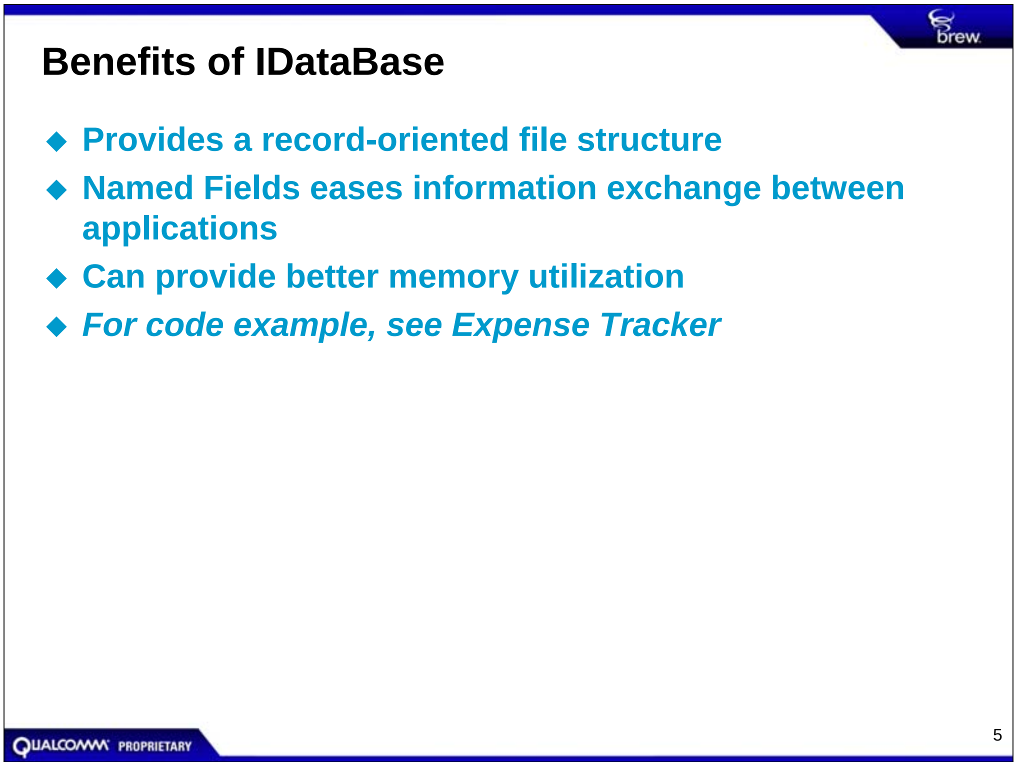


F

Each field of a record contains its name, data type, a pointer to its contents, and the length of the contents. The record access functions in the IDBRecord Interface operate on the current field of the record. IDBRECORD_Reset() makes the first field of the record the current one, and IDBRECORD_NextField() advances the current field to next field in the record. IDBRECORD_NextField() also returns the name, data type, and length (but not the contents) of the new current field, or an end-of-record indication when all the fields of the record have been enumerated. IDBRECORD_GetField() returns a pointer to the contents of the current field and also returns the field's name, data type, and length.

The IDBRecord Interface provides some simpler functions that can be used to access a field's contents when the data type of the contents is already known:

- IDBRECORD_GetFieldWord() retrieves the contents of the current field when its type is word.
- IDBRECORD_GetFieldWord() retrieves the contents of the current field when its type is double-word.
- IDBRECORD_GetFieldString() retrieves the contents of the current field when its type is character string.



Benefits of IDataBase

- Provides a record-oriented file structure
- Named Fields eases information exchange between applications
- Can provide better memory utilization
- *For code example, see Expense Tracker*

QUALCOMM PROPRIETARY

5